

Quantum hashing based on symmetric groups

M. Ziatdinov

Abstract

The notion of quantum hashing formalized by F. Ablyayev and A. Vasiliev in 2013. F. Ablyayev and M. Ablyayev in 2014 introduced the notion of quantum hash generator which is convenient technical tool for constructing quantum hash functions. M. Ziatdinov in 2014 presented group approach for constructing quantum hash functions. All these mentioned above results present constructions of quantum hash functions based on abelian groups.

This paper continue the research on quantum hashing. Our approach allows us to construct quantum hash function working on any (finite) group. Also our approach allows us to construct quantum hash functions based on classical hash function from NC¹.

Keywords: quantum hashing, quantum hashing on groups, symmetric groups

1 Introduction

H. Buhrman et al. [6] introduce the notion of quantum fingerprinting. Quantum fingerprinting based on binary error correcting codes. Later F. Ablyayev and A. Vasiliev in [3] offer another (non binary) version of quantum fingerprinting. F. Ablyayev and A. Vasiliev [4] defined notion of quantum hash-function and showed that quantum fingerprinting is a specific case of quantum hashing.

In [1] construction of Buhrman et al.[6] and Ablyayev-Vasiliev's construction [4] are generalized. It is shown that both approaches can be viewed as composition of so called “quantum generator” and (classical) universal hash function.

In [7] we offered a group approach to fingerprinting. We showed that instead of abelian group \mathbb{Z}_m with $m > 0$ [4] we can use arbitraey abelian group. These constructions use specific so called “good” set of automor-

phisms. However, examples of such “good” sets (and, hence the quantum hash functions) were found only for abelian groups.

In this paper we offer “good” set of automorphisms for symmetric group, and construct quantum hash function based on any finite group. This approach allows us to construct quantum hash functions based on classical functions from NC¹. We also discuss the procedure of finding “good” set of automorphisms.

2 Previous work

We start with recalling basic definitions that we will need in paper.

We will consider functions $h : \{0, 1\}^n \rightarrow G$, where G is a group.

Let us choose a set of automorphisms \mathcal{K} from group of all automorphisms $\text{Aut}(G)$:

$$k_i \in \mathcal{K} \subseteq \text{Aut}(G), \quad 1 \leq i \leq T, |\mathcal{K}| = T \quad (1)$$

We will use notation $k\{g\}$ for image of g under automorphism k .

Let us also choose a homomorphism f from group G to a group of all unitary transformations of m qubits.

Let us recall definitions and theorems from [4] and [7]

Quantum hash function is defined as follows.

Definition 1. $|\Psi(w)\rangle$ is a quantum hash function if it maps n -bit message w from $\{0, 1\}^n$ to m qubits and resulting vectors are nearly orthogonal: $\forall w, w' \in \{0, 1\}^n (|\langle \Psi(w) | \Psi(w') \rangle| < \epsilon)$ for some $\epsilon \in (0, 1)$.

We call set K_{good} of elements of chosen \mathcal{K} “good” set if for each non-unit group element g and some starting state $|\psi_0\rangle$:

$$\forall g \in G, g \neq e : \frac{1}{|K_{\text{good}}|^2} \left| \sum_{k \in K_{\text{good}}} \langle \psi_0 | f(k\{g\}) | \psi_0 \rangle \right|^2 < \epsilon \quad (2)$$

In [7] it was proved that

Theorem 1. *If (3) holds, then “good” set exists and can be constructed by choosing d times element from \mathcal{K} at random, and $d = \frac{2}{\epsilon} \ln |G|$*

$$\forall g \in G, g \neq e : \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \langle \psi_0 | f(k\{g\}) | \psi_0 \rangle = 0, \quad (3)$$

so, if (3) holds, there exists quantum hash function for arbitrary small ϵ (however, “good” set size d and therefore qubit count m will grow)

We will say “quantum hash function works for group G ” or simply “quantum hash function for group G ” if it has form

$$|\Psi_{h,G,K,f,m,|\Psi_0\rangle}(x)\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \left(|j\rangle \otimes f(k_j\{h(x)\}) | \psi_0 \rangle \right), \quad (4)$$

where h is classical hash function mapping X^n to group G , $K = \{k_0, \dots, k_{t-1}\}$ is “good” set of automorphisms and f is homomorphism from G to space $[(\mathcal{H}^2)^{\otimes m} \rightarrow (\mathcal{H}^2)^{\otimes m}]$.

It was also proven that

Theorem 2. *If for group G “good” set of automorphisms exist, then quantum hash function for group G exist.*

3 Quantum hash function working on symmetric group

Theorem 3. *There exists a quantum hash function $|\Psi_{h,S_n,K,f,\log n}\rangle$ working on symmetric group.*

Specifically, f is standard symmetric group representation in a space of n dimensions and K is a set of all automorphisms acting by conjugation to cyclic shift.

Proof. Theorems 1 and 2 state that if there exists a homomorphism f , a set \mathcal{K} of automorphisms of G such that

$$\frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \langle \psi_0 | f(k\{g\}) | \psi_0 \rangle = 0, \quad (5)$$

then $\Psi_{h,G,K,f,m}$ is a quantum hash function.

In our case, f is a standard symmetric group representation in a space of n dimensions with group S_n acting by coordinates permutation.

Let \mathcal{K} be the set of all (inner) automorphisms that has form:

$$\mathcal{K} = \{g_\sigma : \sigma \text{ is a cyclic shift}\}, \quad g_\sigma(\tau) = \sigma\tau\sigma^{-1} \quad (6)$$

Let $|\psi_0\rangle$ be some vector $c_1|1\rangle + c_2|2\rangle + \dots + c_n|n\rangle$, such that:

$$\sum_{i=1}^n c_i = 0 \quad (7)$$

Image of $|\psi_0\rangle$ under $f(g_\tau\{\sigma\})$ for any σ and $\tau \in \mathcal{K}$ is

$$f(g_\tau\{\sigma\}) = c_{\sigma(1+k)-k}|1\rangle + \dots + c_{\sigma(n+k)-k}|n\rangle, \quad (8)$$

where τ is cyclic shift to k and addition and subtraction in indices are modulo n .

So, if we sum this for all automorphisms $\tau \in \mathcal{K}$ we get:

$$\sum_{g_\tau \in \mathcal{K}} \langle \psi_0 | f(g_\tau\{\sigma\}) | \psi_0 \rangle = \sum_{k=0}^n \sum_{i=0}^n c_i c_{\sigma(i+k)-k} = \sum_{i=0}^n c_i \sum_{k=0}^n c_{\sigma(i+k)-k}. \quad (9)$$

We substituted $f(g_\tau\{\sigma\})|\psi_0\rangle$ with its value from 8.

We can observe that $\sigma(i+k) - k$ runs over all integers from 1 to n . So we can rewrite as follows:

$$\sum_{g_\tau \in \mathcal{K}} \langle \psi_0 | f(g_\tau\{\sigma\}) | \psi_0 \rangle = \sum_{i=0}^n c_i \sum_{j=0}^n c_j = 0. \quad (10)$$

We use equation (9) and definition (7) of ψ_0 .

Equation (10) is equivalent to (5), so theorems 1 and 2 can be applied, and quantum hash function for S_n exist. \square

Please note that this proof does not apply to A_5 representation from paper [2] and we cannot use their representation and approach of this article to define quantum hash functions based on NC^1 functions. In the section 4 we use another representation.

In [7] it was shown that if we find a set \mathcal{K} satisfying equation (2), we can construct a “good” set with probability of $\frac{1}{|G|}$ by repeatedly ($d = \frac{2}{\epsilon} \ln |G|$ times) randomly choosing an element from \mathcal{K} .

4 Applications

We can use defined quantum hash function working on symmetric group to construct other quantum hash functions. One way of such construction is defined in [7]: we construct a hash function working on (direct) product of groups. We present another way.

Lemma 1. *Let G be a finite group, group $G' \triangleleft G$ be its subgroup, and $|\Psi_{h,G,K,f,m}\rangle$ be a quantum hash function working on it.*

Then we can define a quantum hash function working on G' .

Proof. We can define h' to be a restriction of h on G' .

Then $|\Psi_{h',G',K,f,m}\rangle$ is a quantum hash function.

Let us consider square of scalar product of quantum hash function values on different inputs.

$$|\langle \Psi_{h',G',K,f,m}(x) | \Psi_{h',G',K,f,m}(x) \rangle|^2 = |\langle \Psi_{h,G,K,f,m}(x) | \Psi_{h,G,K,f,m}(x) \rangle|^2 < \epsilon$$

We use that $G' \triangleleft G$ and that h' is a restriction of h on G' . \square

Of course, such way is inefficient for small finite subgroups of S_n , but it works for non-abelian groups.

We can use our approach to construct quantum hash functions based on classical hash functions in NC^1 .

Let h be a hash function that can be computed by NC^1 circuit. We can now use theorem 3 to obtain a quantum hash function based on it as follows.

We can convert circuit to width-5 polynomial-size branching program and represent it as permutation branching program [5]. Then we compute quantum hash function based on h as follows. For each input symbol we simultaneously apply required permutation in all subspaces (under different automorphisms as described in theorem 3).

References

- [1] F. Ablayev, M. Ablayev. Quantum Hashing via Classical ϵ -universal Hashing Constructions. arXiv:1404.1503v2 [quant-ph] 2014
- [2] F. Ablayev, C. Moore, and C. Pollett. Quantum and stochastic branching programs of bounded width, *Proceedings of the International Colloquium on Automata, Languages, and Programming (ICALP), 2002*. arXiv:quant-ph/0201139.
- [3] F. Ablayev, A. Vasiliev. Algorithms for quantum branching programs based on fingerprinting. *Electronic Proceedings in Theoretical Computer Science* 9: 1–11, 2009.
- [4] F. Ablayev, A. Vasiliev. Cryptographic quantum hashing. *Laser Physics Letters* 11.2, 2014.
- [5] D. A. M. Barrington. Bounded-width polynomial-size branching programs can recognize exactly those languages in NC_1 , *Journal of Computer and System Sciences* 38:150-164, 1989.
- [6] H. Buhrman, R. Cleve, J. Watrous, R. de Wolf. Quantum fingerprinting. *Physical Review Letters*, 87(16), 2001.
- [7] M. Ziatdinov. Quantum hashing. Group approach. http://shelly.kpfu.ru/e-ksu/docs/F1221792420/hash_8_ljm_en_draft.pdf